TAMU LARSYS III.1

AN IBM 370 VERSION

CONVERSION OF LARSYS III.1 TO AN IBM 370 COMPUTER

Final Project Report

Principal Investigator:
Glen N. Williams, Ph.D., P.E.

Co-Investigators:
John Leggett
Glen A. Hascall

Contract NAS9-14514

by

Texas A&M University
College Station, Texas

for

Earth Observations Division
National Aeronautics and Space Administration
Johnson Space Center
Houston, Texas

ABSTRACT

In March 1975, the Texas A&M University Data Processing Center was awarded Contract Number NAS9-14514 from the National Aeronautics and Space Administration - Johnson Space Center, entitled "Conversion of LARSYS III.1 to an IBM 370." The contractual effort was successfully completed in February 1976.

LARSYS, a software system for processing multispectral aircraft or satellite data, was designed and written at the Laboratory for Applications of Remote Sensing at Purdue University. This system, being implemented on an IBM 360/67 computer utilizing the Cambridge Monitor System, is of an interactive nature.

TAMU LARSYS maintains the essential capabilities of Purdue's LARSYS. The machine configuration for which it has been converted is an IBM-compatible Amdahl 470V/6 computer utilizing the Time Sharing Option of the currently implemented OS/VS2 Operating System.

Due to TSO limitations, the NASA-JSC deliverable TAMU LARSYS is comprised of two parts. Part One is a TSO Control Card Checker for LARSYS control cards, and Part Two is a batch version of LARSYS. Used together, they afford most of

the capabilities of the original LARSYS III.1.

Additionally, two programs have been written by TAMU to support LARSYS processing. The first is an ERTS-to-MIST Conversion program used to convert ERTS data to the LARSYS input form, the MIST tape. The second is a System Runtable code which maintains tape/file location information for the MIST data sets.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

LIST OF TABLES

LIST OF FIGURES

TAMU LARSYS SYSTEM PROGRAMMER'S MANUAL

Introduction

LARSYS is a software system which aids the user in the reduction and display of multispectral data captured by high altitude aircraft or satellite. LARSYS was designed and written by the personnel of the Laboratory for Applications of Remote Sensing (LARS) at Purdue University. At Purdue LARSYS runs on an IBM 360/67 Central Processing Unit (CPU) under the CP-67 Control System and the Cambridge Monitor System (CMS) Operating System. This implementation allows LARSYS to be presented to the user as a time-shared interactive virtual system. In this type of environment, the user sits at his terminal answering system responses and getting results as if he had the whole CPU to himself while, in fact, he is sharing the system resources with a number of other users.

In March 1975, the Texas A&M University (TAMU) Data Processing Center was awarded a contract from the National Aeronautics and Space Administration - Johnson Space Center, to convert the current version of LARSYS (LARSYS III.1) to an operational status on the IBM 370 series of computers.

The TAMU Data Center currently runs an Amdahl 470V/6 CPU, which is totally compatible with the IBM 360/370 lines of computers. The Amdahl is controlled by the IBM OS/VS2 Operating System. While some general purpose interactive computation is available under the Time Sharing Option (TSO) of OS/VS2, the TAMU Data Center is primarily a "batch" shop. Thus TSO working regions (in memory) are limited in size and priority. In addition, the big drawback to TSO is that it is very limited in its Input/Output support compared to CMS. TSO only supports I/O to direct access devices, not tapes or unit record equipment. This places a serious limitation on the LARSYS system, since LARSYS was designed for tapes and unit record equipment as well as direct access devices. For these reasons it was decided to convert LARSYS III.1 into a two part system. Part One is an interactive Control Card Checker implemented through TSO, and Part Two is a batch version of the LARSYS III.1 system.

The Control Card Checker is designed to allow users to interactively (at a terminal) run LARSYS control card decks through the system and perform only error checking. The card images are checked for syntax and ambiguous meaning using the same routines which LARSYS uses in the '-CHECKOUT' mode of operation. As errors are found, the user is notified

at his terminal and is asked to change the card. The replacement cards are then stored along with those with no errors to form a corrected deck. This corrected deck may then be used with the batch version of LARSYS.

It is envisioned that the user will run LARSYS jobs in both the interactive and batch steps. Although not mandatory, this procedure is recommended, since batch processing will terminate if an error is detected on a control card. With use of the Control Card Checker, however, this limitation should not be encountered very often.

The batch version of LARSYS includes virtually the entire LARSYS III.1 system, with all commands and functions implemented which are available at Purdue. Although the system is complete, the routines that dealt with Purdue's special Display Device were not changed at TAMU due to lack of this equipment. Thus, this processor is not considered operational.

The major changes to LARSYS III.1 were in the interactive requests to the user and in the tape operations of the system. All interactive requests to the user have been preceded by a call to a FORTRAN routine (INTREQ) which outputs a message indicating that an interactive request was made and the name of the routine making the request. After

this message is output, execution is terminated. The Tape Operations Program (TAPOP) in LARSYS III.1 was completely re-written since this program was heavily CMS dependent. The new TAPOP handles all of the old TAPOP Entry Points and operands. It also includes two subroutines and various CMS functions from LARSYS III.1. The new TAPOP is totally transparent to the FORTRAN processing routines.

All changes made to the FORTRAN and Assembly Language programs were documented by commenting the original statements and inserting the new statements with the letters 'TAM' in columns 73-75 and a sequence number in columns 76-80. This allows the changes to the programs to be easily recognized when looking at a listing of the program.

The remainder of this Volume is divided into seven sections. Section One presents detailed instructions for the generation of the TAMU LARSYS system on another IBM-compatible computer. Sections Two and Three discuss the LARSYS Processing Level files and System Information files, respectively. Sections Four, Five, and Six are comprised of information necessary in the testing and validation of the LARSYS system. Finally, Section Seven provides a complete listing of all data sets used in the TAMU LARSYS system.

TAMU LARSYS Implementation

Introduction

The Texas A&M University version of LARSYS III.1 is implemented on an AMDAHL 470V/6 CPU running under the IBM Operating System VS2, Release 1.7, with the Time Sharing Option (TSO). As described in the TAMU User's Manual, there are two parts comprising the system implemented at TAMU. One is the TSO Control Card Checker which runs interactively under TSO; the other is a batch version of the complete LARSYS system.

The LARSYS installation procedure consists of unloading the compiled and linked load modules from the TAMU Distribution Tape and allocating and copying the required run time data sets. If for some reason one or more of these load modules do not execute properly, then the complete regeneration process should be followed for the required load module(s).

All of the source program modules necessary to generate the TAMU version of LARSYS, in addition to the source code for the ERTS-to-MIST conversion program (EMCNV) and the System Runtable builder (RUNTABLE), are included on the TAMU Distribution Tape, the contents of which are shown in Table

1. The LARSYS test card decks and MIST data files are not included, as they are available on the LARS Distribution Tapes, assumed to be on-site.

LARSYS Installation Procedure

Step 1: Punch the JCL card image file from file 1 of the TAMU Distribution Tape. JCL similar to the following can be used:

```
//PUNCH  EXEC  PGM=IEBGENER
//SYSPRINT  DD  SYSOUT=A
//SYSIN  DD  DUMMY
//SYSUT1  DD  UNIT=2400,VOL=SER=TAMLRS,DISP=(OLD,PASS),
//        LABEL=(1,SL),DSN=JCLCARDS,DCB=(BLKSIZE=800,LRECL=80,
//        RECFM=FB)
//SYSUT2  DD  SYSOUT=B,DCB=(RECFM=F,BLKSIZE=80)
```

Step 2: Using the JCL cards from Step 1 marked "SYSTEM RUNTIME ALLOCATION," allocate the data sets necessary to run the LARSYS system. Note that the JCL is in the form of an in-line Procedure with the variable parameter "VOLUME." If the TSO part of the system is to be used, this parameter should be pointed to an on-line disk pack; otherwise, it may be pointed to a mountable pack.

The JCL should be checked carefully to insure that further tailoring is not necessary for installation at the host site.

Step 3: Using the JCL from Step 1 marked "UNLOAD LARSYS LOAD MODULES," unload the library on file 2 of the TAMU Distribution Tape.

Summary: At this point the system is installed and ready for testing. The procedures in the JCL from Step 1 marked "RUN BATCH LARSYS," "RUNTABLE BUILDER," and "ERTS-MIST CONVERSION" may be used as in-line Procedures or stored in a local Procedure Library. For the TSO Control Card Checker, the JCL marked "TSO LOGON PROC" must be put in the local TSO LOGON Procedure Library and its name added to the user's UADS. Also the two CLISTs, marked "RUN TSO LARSYS" and "SUBMIT LARSYS," must be placed in a common CLIST Library.

The installed system should now be tested according to the instructions in this manual under the section heading of Testing Procedures.

LARSYS System Regeneration

In the event that one or more of the load modules do not execute properly, then it (they) must be regenerated as per

the following.

The card images punched in Step 1 above contain three JCL command streams, one each for TAMU batch LARSYS, the TSO Control Card Checker, and the TAMU-supplied utilities, RUNTABLE and EMCNV. These are appropriately marked with "REGENERATE BATCH," "REGENERATE TSO," and "REGENERATE UTILITIES." Each JCL command stream contains all of the job steps necessary to compile and link-edit that part of the system.

The major logical Phases are:

Phase 1. Allocate libraries for source modules and load modules.

Phase 2. Unload the appropriate source file from the TAMU Distribution Tape.

Phase 3. Compile FORTRAN modules.

Phase 4. Assemble Assembly Language modules.

NOTE: If the load module in error is the batch LARSYS module, the error is probably due to Operating System incompatibilities between TAMU and the host installation as concerns the Tape Operations subroutine, TAPOP. Therefore, it is suggested that the following steps be taken to insure that TAPOP is executing properly.

Step 1. Punch the source code for the TOPTST and TAPOP modules.

Step 2. Refer to the section in the manual entitled Tape Operations Test Program, noting the necessary changes.

Step 3. Assemble the TAPOP module, compile the TOPTST module, and link these modules together with modules FORTO, ERPRNT, RTMAIN, MSG1, and ERMNAM. JCL similar to the following will accomplish this task.

```
//STEP1 EXEC ASMXC,MAC1='LIBRARY',PARM.ASM='LOAD,NODECK'
//ASM.SYSIN DD *
   --TAPOP SOURCE CODE--
//STEP2 EXEC FORTGCLG,REGION.GO=128K
//FORT.SYSIN DD *
   --TOPTST SOURCE CODE--
//LKED.TAPOP DD DSN=LARSYS.SYSGEN.BATLOAD,DISP=SHR
//LKED.SYSIN DD *
 INCLUDE TAPOP(FORTO,ERPRNT,MSG1,ERMNAM,RTMAIN)
//GO.FT08F001  DISP=SHR,DSN=LARSYS.SYSRUN.ERRORS.CNTL
//GO.FT11F001 DD UNIT=(TAPE9,,DEFER),VOL=SER=DUMMY1,LABEL=(1,NL),
//    DCB=(RECFM=VS,BLKSIZE=1500,DEN=3,BUFNO=1),DISP=(OLD,KEEP)
//GO.FT12F001 DD UNIT=(TAPE9,,DEFER),VOL=SER=DUMMY2,LABEL=(1,NL),
//    DCB=(RECFM=VS,BLKSIZE=1500,DEN=3,BUFNO=1),DISP=(OLD,KEEP)
//GO.FT13F001 DD DUMMY
//GO.FT16F001 DD SYSOUT=A
//GO.SYSUDUMP DD SYSOUT=A
//GO.SYSIN DD *
   --DATA--
/*
```

The 'MAC1' parameter on the 'STEP1 EXEC' card points to a library containing the DSECT,IEFJFCBN. Also note the 'BUFNO' and 'VOL=SER' parameters on the 'GO.FT11F001 DD' and 'GO.FT12F001 DD' cards, as these must be specified as shown.

Step 4. Follow the instructions in the LARSYS Test Procedures Manual for TOPTST execution.

Phase 5. Scratch old load module from the library created in Step 2 of the LARSYS Installation Procedure.

Phase 6. Compress load library.

Phase 7. Link-edit individual modules.

It is suggested that the compile/link-edit steps in Phases 3 and 4 above be run in groups of five or six at a time, as system queue space limitations usually preclude loading the entire JCL command stream at once.

Contents of TAMU Distribution Tape

General. The tape distributed by Texas A&M University contains the JCL and load modules necessary to run the TSO Control Card Checker, TAMU batch LARSYS, and the TAMU-supplied utility programs, RUNTABLE and EMCNV. In addition, JCL and source code is provided to regenerate any of the above load modules. The characteristics of the tape itself are: Standard Label, VOL=SER=TAMLRS, 9-Track, 1600 BPI.

Table 1. TAMU Distribution Tape Structure

File	Contains
1	Sequential file containing the JCL necessary to install, run, and regenerate each of the parts of TAMU LARSYS and the utilities. Parameters are: BLKSIZE=800, LRECL=80,DSNAME=JCLCARDS.
2	Unloaded PDS (Partitioned Data Set) containing four load modules (TSOLARS, BATLARS, RUNTABLE, EMCNV). Unloaded by IEBCOPY,DSNAME=LARSYS.LOADLIB.
3	Sequential file containing Error messages used during LARSYS execution. Parameters are: BLKSIZE=800, LRECL=80,DSNAME=LARSYS.ERRORS.
4	Sequential file containing the "REFERENCE" data for the Control commands. Parameters are: BLKSIZE=800, LRECL=80,DSNAME=LARSYS.CONTROL.
5	Sequential file containing the "REFERENCE" data for the Initialization commands. Parameters are: BLKSIZE=800, LRECL=80,DSNAME=LARSYS.INITIAL.
6	Sequential file containing the "REFERENCE" data for the Processor commands. Parameters are: BLKSIZE=800, LRECL=80,DSNAME=LARSYS.PROCESS.
7	Sequential file containing the LARSYS INDEX file. Parameters are: BLKSIZE=800,LRECL=80,DSNAME=LARSYS. INDEX.

Table 1. TAMU Distribution Tape Structure (continued)

File	Contains
	NOTE: Files 3-7 are copied to disk at system allocation and initialization time.
8	Unloaded PDS containing all of the source code necessary to generate TAMU batch LARSYS. Unloaded by IEBCOPY,DSNAME=LARSYS.BATSORC.
9	Unloaded PDS containing all of the source code necessary to generate the TSO Control Card Checker. Unloaded by IEBCOPY,DSNAME=LARSYS.TSOSORC.
10	Unloaded PDS containing all of the source code necessary to generate RUNTABLE and EMCNV. Unloaded by IEBCOPY,DSNAME=LARSYS.UTILSORC.

LARSYS Processing Level Files

The LARSYS processing level files, in general, are maintained by the LARSYS system itself. The two exceptions to this rule are the Error Message file and the Runtable file. The Error Message file was modified at TAMU and is contained on the TAMU Distribution Tape. This file should not need to be modified for the current version of LARSYS. The Runtable file, however, contains pointers to information concerning the current Multispectral Image Storage Tape (MIST) Runs which are available for processing. Therefore, it should be updated each time an ERTS data set is converted to MIST form. The remainder of this section is devoted to the System Runtable file.

The System Runtable

The System Runtable is a direct access data set which links a MIST Run number to the tape and file on which the Run resides. This data set is essential since almost every LARSYS function uses it. The data set consists of 1340 800-byte records. The first 10 records are the Directory records and the last 1330 records are MIST Tape Header records. Each Directory record contains 133 6-byte entries. The first 4

bytes of each entry contain the Run number and the last 2 bytes contain the relative record number of the Tape Header record for the Run.

A program to build and maintain the System Runtable was written at Texas A&M University, since no such program is included in the standard LARSYS Documentation. The program, RUNTABLE, includes six functions. These functions are delineated in the following table.

Table 2. System Runtable Maintenance Functions

Function Keyword	Function Performed	Operands
INIT	Initializes the System Runtable	None
ADD	Adds Runs to the System Runtable	Run, Tape, File
DEL	Deletes Runs from the System Runtable	Run
REPL	Replaces Header records in the System Runtable	Run, Tape, File
COMP	Compresses the System Runtable by removing all deleted entries	None
STAT	Gives statistics on the System Runtable such as number of deleted entries, etc.	None

Input to RUNTABLE is via function selector cards. Each card specifies a function keyword in columns 1-4. The operands, if any, are keyword operands and may be specified in

any sequence and occur anywhere on the card beyond column 4. Commas or blanks may delimit the operands.

RUNTABLE recognizes three operands: Run, Tape and File. These operands are specified as follows: RUN=XXXXXXXX, where the X's represent an 8 digit MIST Run number; TAPE=YYYYYY, where the Y's represent a 6 digit tape serial number; and FILE=ZZ, where the Z's represent a 2 digit file number. Some examples of valid function selector cards are:

```
ADD     RUN=12345678   TAPE=123456   FILE=12
REPL     FILE = 01,TAPE=654321, R U N =876 54 321
COMP
DEL     RUN=71 05 47 00
```

The INITIALIZATION function should be requested only once, since this function indiscriminantly sets the whole System Runtable to binary zeroes. The INITIALIZATION function is used internally as Phase II of the COMPRESS function, therefore the INITIALIZATION function code must remain in RUNTABLE. RUNTABLE uses two DSECTS to map Control Blocks. These DSECTS are IEFJFCBN and DCBD and therefore must be accessible to the Assembler when assembling RUNTABLE.

RUNTABLE requires one tape to add or replace Runs in the System Runtable. Execution is expedited if MIST tape requests are grouped by tape number and then sequentially by file number. This alleviates unneccessary dismounting and

mounting of tapes and allows the mounted tape to move in a sequential fashion.

As an End-of-Job procedure, RUNTABLE calls a FORTRAN subroutine (RUNREF) that updates the Runtable Reference file for LARSYS processing. This subroutine requires only the revision date as a parameter. The last card input to RUNTABLE should have the revision date in columns 1 through 8 in the form MM/DD/YY. If this card is omitted, a warning message is output and the Runtable Reference file is not updated. This card, though, will act as a convenient trailer card to the function selector cards provided for RUNTABLE processing. An example of a System Runtable Update is shown in Figure 1.

For more information on the System Runtable, see LARSYS System Manual, June 1, 1973, pages 5-28 and 5-29.

Figure 1. Example of a System Runtable Update

```
** RUNTABLE UTILITY **

*INITIALIZATION FUNCTION REQUESTED
-INITIALIZATION FUNCTION COMPLETED

*ADD FUNCTION REQUESTED
-ADD FUNCTION COMPLETED

*ADD FUNCTION REQUESTED
-ADD FUNCTION COMPLETED

*REPLACE FUNCTION REQUESTED
-REPLACE FUNCTION COMPLETED

*DELETE FUNCTION REQUESTED
-DELETE FUNCTION COMPLETED

*STATISTICS FUNCTION REQUESTED
NUMBER OF ACTIVE ENTRIES = 0001
NUMBER OF INACTIVE ENTRIES = 0001
TOTAL RECORDS USED = 0002 OUT OF POSSIBLE 1330
-STATISTICS FUNCTION COMPLETED

*COMPRESS FUNCTION REQUESTED
*INITIALIZATION FUNCTION REQUESTED
-INITIALIZATION FUNCTION COMPLETED
-COMPRESS FUNCTION COMPLETED

*STATISTICS FUNCTION REQUESTED
NUMBER OF ACTIVE ENTRIES = 0001
NUMBER OF INACTIVE ENTRIES = 0000
TOTAL RECORDS USED = 0001 OUT OF POSSIBLE 1330
-STATISTICS FUNCTION COMPLETED

 NO REVISION DATE -- RUNTABLE

    REFERENCE NOT UPDATED

** RUNTABLE UTILITY FINISHED **
```

LARSYS System Information Files

LARSYS System Information files are files that are created and/or maintained by the LARSYS System Programmer as part of the process of updating the LARSYS system. They constitute data for the operation of the LARSYS 'NEWS', 'REFERENCE', and 'LIST' Control commands. All of these files consist of 80 character card images except the Runtable Reference file which consists of 120 character lines. The Runtable Reference file is automatically updated by the RUNTABLE program. The other files may be maintained by a simple program which reads card images and places them on disk.

The LARSYS System Information REFERENCE and LARSYS INDEX files were modified at TAMU and appear on the TAMU Distribution Tape. The modifications to these files deal mainly with indicating that certain instructions (DISCONNECT, etc.) were not supported because the instructions are of an interactive nature.

All of the NEWS files contain installation dependent information and thus are not contained on the TAMU Distribution Tape. Therefore, they should be allocated and defined with the desired contents prior to LARSYS execution.

For more information on LARSYS System Information files see LARSYS System Manual, June 1, 1973, pages 5-68 through 5-70.

Testing Procedures

The following steps must be completed prior to testing TAMU batch LARSYS.

Step 1. Copy the MIST data files from the LARS Distribution Tapes to host installation tapes, changing the first and second words (tape and file number, respectively) of the Header records accordingly. It is suggested that Run 71053001 be copied onto file one of a tape, as this Run is used extensively in the test procedures. Also, Run 66000600 should be copied onto file one of a tape as this is required by TOPTST.

Step 2. Build the System Runtable according to the instructions set forth under the section heading of LARSYS Processing Level Files in this manual.

Step 3. Punch the test decks from the LARS Distribution Tape.

Testing may now begin as shown in the LARS Test Procedure Manuals.

Test Output Differences Between LARS and TAMU Batch LARSYS

There were very few differences between TAMU test output and the LARS Test Volumes. The differences that were noted, however, are summarized below according to the Test Volume and particular test in which they occurred. All interactive tests were modified to remove the interactivity, then the test was performed. No differences were noted in any of these tests.

Volume I

Tape Handling Test

Page 2-18: Insert between lines 25 and 26

```
25  TAPOP--TYPE IN DESIRED PHASE
    QUERY RESPONSE = 10.
    QUERY RESPONSE = 0.
26  PHASE IX TEST OK
```

Control Card Checkout

Page 4-16: Change line 42

```
From E465  NO RUN NUMBER WAS GIVEN-REQUEST CANCELLED...
           (PICRDR)
To   E520  NO RUN NUMBER GIVEN-TYPE THE "DISPLAY CARD"...
           (IMARDR)
   - Add Immediately -
     DISPLAY RUN (XXXXXXXX)
```

Volume I

HISTST1

Pages 5-136 and 5-137:

The test deck furnished by Purdue requests lines 12 to 159 to be histogrammed (page 5-136, line 2) but the test output (page 5-137, lines 14, 16, 18) shows output for lines 12 to 84. Also, a punched card deck is produced from this test. The punched deck does not match the deck supplied by Purdue due to the different lines histogrammed.

Volume II

RUNTST2

Page 5-223:

CO = -0.5E5 For all Channels (1, 3, 5, 7, 9, 11)

RUNTST3

Page 5-237:

CO = -0.5E5 For Channel 1

CLUTST1

Page 5-250: Delete lines 52 through 64.

Message I0172 does not appear on the test output.

CLUTST2

Page 2-278:

The punched deck output for this test that was sup-

Volume II

CLUTST2

plied by Purdue is 135 cards short. The deck produced at Texas A&M matched Purdue's punched deck up to the end of the Purdue deck.

Volume III

There were no discrepancies noted in Volume III.

Volume IV

PRIGEN

Page 5-871: Delete lines 5 and 6

```
5  I0002 TAPE 1007 HAS BEEN REQUESTED
       ON UNIT 0181 (MOUNT)

6  I0003 TAPE READY...EXECUTION CONTINUING (MOUNT)
```

This last difference may occur in other places due solely to the positioning of the Test MIST Runs on the local tapes. This merely reflects that the MIST Runs were stored on tape in different order at TAMU than at LARS.

Tape Operations Test Program

TOPTST is a program supplied by Purdue to test the Tape Operations subroutine (TAPOP). The changes to TOPTST were minimal and were made in the same fashion as the changes to the LARSYS III.1 FORTRAN programs. That is, original statements were commented and replacement statements, with 'TAM' in columns 73-75 and a sequence number in columns 76-80, were inserted. The changes are summarized in the following table.

Table 3. Changes to Tape Operations Test Program

Statement Sequence Number Cols. 76-80	Reason for Change
155	To make input come from a card reader, and thus be batch usable.
435 and 7295	Allows Phase X to be run immediately after Phase IX.
632 through 645 and 5664 through 5668	Reflects the change in MOUNT to handle GTUNIT and RINGIN responsibilities as well as mounting the required tape.
5040 and 5045	Added length of record written (when End-of-Tape mark was encountered) to output list.
7215, 7235 and 7254 through 7258	Addition of a test for the QUERY REAL UNIT feature that has been added to TAPOP (replaces CMS QUERY REAL UNIT feature).

Table 3. Changes to Tape Operations Test Program (continued)

Statement Sequence Number Cols. 76-80	Reason for Change
7220, 7245, 7252 and 7375	The function of TCLOSE is now incorporated in either of two Entry Points - TOPRU or DETACH (Both close the Data Control Block for the requested tape).

The new TOPTST is included as one of the FORTRAN routines that is supplied in modified source form. TOPTST may be punched from the source tape and run with the TAPOP module to provide a test of tape operations. One statement in TOPTST will have to be modified. This statement has sequence number 05664. It is a call to Subroutine MOUNT requesting tape ZZ3657. The tape parameter must be changed to request a tape which has MIST Run 66000600 as file one. This will match LARSYS documentation for Phase VIII of the TOPTST program.

TAMU LARSYS Data Sets

Table 4 gives a complete listing of all data sets used by the TAMU LARSYS system.

Table 4. TAMU LARSYS Data Sets

DSRN	Description	DD Statement Parameters
1	STATISTICS File	UNIT=SYSDA,SPACE=(TRK,(5,2)),DCB=(BLKSIZE=80,RECFM=F)
2	HISTOGRAM File	UNIT=SYSDA,SPACE=(TRK,(5,2)),DCB=(BLKSIZE=360,RECFM=F)
3	CLUSTER Scratch File	UNIT=SYSDA,SPACE=(TRK,(5,2)),DCB=(BLKSIZE=800,RECFM=VS)
4	PRINT RESULTS Scratch File	UNIT=SYSDA,SPACE=(TRK,(5,2)),DCB=(BLKSIZE=121,RECFM=F)
5	Card Reader	DDNAME=SYSIN
6	Printer	SYSOUT=A
7	Punch	SYSOUT=B
8	Error Message File	DISP=SHR,DSN=LARSYS.SYSRUN.ERRORS.CNTL, DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
9	System Runtable File	DISP=SHR,DSN=LARSYS.SYSRUN.RUNTABLE.DATA, DCB=(BLKSIZE=800,RECFM=F)
10	Training and Test Fields File	UNIT=SYSDA,SPACE=(TRK,(5,2)),DCB=(BLKSIZE=800,RECFM=VS)
11	CLASSIFICATION Results Tape, SEPARABILITY Scratch Tape, and DUPLICATERUN Output Tape	UNIT=(TAPE9,,DEFER),VOL=SER=DUMMY1, LABEL=(1,NL),DISP=(OLD,KEEP),DCB=(RECFM=VS,BLKSIZE=1500,DEN=3,BUFNO=1)

Table 4. TAMU LARSYS Data Sets (continued)

DSRN	Description	DD Statement Parameters
12	MIST Tape and COPYRESULTS Output Tape	UNIT=(TAPE9,,DEFER),VOL=SER=DUMMY2, LABEL=(1,NL),DISP=(OLD,KEEP),DCB= (RECFM=VS,BLKSIZE=1500,DEN=3,BUFNO=1)
13	TRANSFERDATA Output Tape	UNIT=(TAPE9,,DEFER),VOL=SER=DUMMY3, LABEL=(1,NL),DISP=(OLD,KEEP),DCB= (RECFM=F,BLKSIZE=80,DEN=3,BUFNO=1)
15	Terminal Keyboard	DUMMY (Non-existent)
16	Terminal Typewriter	SYSOUT=A
17	CLASSIFICATION Results File	UNIT=SYSDA,SPACE=(TRK,(5,2)),DCB= (BLKSIZE=800,RECFM=VS)
18	SEPARABILITY Scratch File	UNIT=SYSDA,SPACE=(TRK,(5,2)),DCB= (BLKSIZE=800,RECFM=VS)
19	Field Boundaries File	UNIT=SYSDA,SPACE=(TRK,(5,2)),DCB= (BLKSIZE=800,RECFM=VS)
20	CLASSIFYPOINTS Scratch File	UNIT=SYSDA,SPACE=(TRK,(5,2)),DCB= (BLKSIZE=800,RECFM=VS)
21	Permanent STATISTICS File	DISP=SHR,DSN=LARSYS.SYSRUN.PSDATA, DCB=(RECFM=F,BLKSIZE=80),SPACE=(TRK, (10,2),,CONTIG)
22	Permanent HISTOGRAM File	DISP=SHR,DSN=LARSYS.SYSRUN.PHDATA, DCB=(RECFM=F,BLKSIZE=360),SPACE=(TRK, (10,2),,CONTIG)
23	LARSYS INDEX File	DISP=SHR,DSN=LARSYS.SYSRUN.INDEX, DCB=(RECFM=FB,BLKSIZE=80,LRECL=80), SPACE=(TRK,(3,1),,CONTIG)
24	PRIORITY NEWS File	DISP=SHR,DSN=LARSYS.SYSRUN.NEWS.PRIORITY, DCB=(RECFM=FB,BLKSIZE=80,LRECL=80), SPACE=(TRK,(3,1),,CONTIG)
25	SYSTEM NEWS File	DISP=SHR,DSN=LARSYS.SYSRUN.NEWS.SYSTEM, DCB=(RECFM=FB,BLKSIZE=80,LRECL=80), SPACE=(TRK,(3,1),,CONTIG)

Table 4. TAMU LARSYS Data Sets (continued)

DSRN	Description	DD Statement Parameters
26	SCHEDULE NEWS File	DISP=SHR,DSN=LARSYS.SYSRUN.NEWS.SCHEDULE, DCB=(RECFM=FB,BLKSIZE=80,LRECL=80), SPACE=(TRK,(1,1),,CONTIG)
27	SYSTEM UPDATE NEWS File	DISP=SHR,DSN=LARSYS.SYSRUN.NEWS.UPDATES, DCB=(RECFM=FB,BLKSIZE=80,LRECL=80), SPACE=(TRK,(3,1),,CONTIG)
28	Control Commands Reference File	DISP=SHR,DSN=LARSYS.SYSRUN.REF.CONTROL, DCB=(RECFM=FB,BLKSIZE=800,LRECL=80), SPACE=(TRK,(5,1),,CONTIG)
29	Initialization Commands Reference File	DISP=SHR,DSN=LARSYS.SYSRUN.REF.INITIAL, DCB=(RECFM=FB,BLKSIZE=800,LRECL=80), SPACE=(TRK,(5,1),,CONTIG)
30	Processor Commands Reference File	DISP=SHR,DSN=LARSYS,SYSRUN.REF.PROCESS, DCB=(RECFM=FB,BLKSIZE=800,LRECL=80), SPACE=(TRK,(20,1),,CONTIG)
31	Runtable Reference File	DISP=SHR,DSN=LARSYS.SYSRUN.REF.RUNTABLE, DCB=(RECFM=F,BLKSIZE=120),SPACE=(TRK, (3,1),,CONTIG)
32	Accounting Data File	DISP=MOD,DSN=LARSYS.SYSRUN.ACCTDATA, DCB=(RECFM=F,BLKSIZE=80), SPACE=(TRK,(10,2))

TAMU LARSYS USER'S MANUAL

Introduction

LARSYS is a software system which aids the user in the reduction and display of multispectral data captured by high altitude aircraft or satellite. LARSYS was designed and written by the personnel of the Laboratory for Applications of Remote Sensing (LARS) at Purdue University. At Purdue LARSYS runs on an IBM 360/67 Central Processing Unit (CPU) under the CP-67 Control System and the Cambridge Monitor System (CMS) Operating System. This implementation allows LARSYS to be presented to the user as a time-shared interactive virtual system. In this type of environment, the user sits at his terminal answering system responses and getting results as if he had the whole CPU to himself while, in fact, he is sharing the system resources with a number of other users.

In March 1975, the Texas A&M University (TAMU) Data Processing Center was awarded a contract from the National Aeornautics and Space Administration - Johnson Space Center, to convert the current version of LARSYS (LARSYS III.1) to

an operational status on the IBM 370 series of computers. The TAMU Data Center currently runs an Amdahl 470V/6 CPU, which is totally compatible with the IBM 360/370 lines of computers. The Amdahl is controlled by the IBM OS/VS2 Operating System. While some general purpose interactive computation is available under the Time Sharing Option (TSO) of OS/VS2, the TAMU Data Center is primarily a "batch" shop. Thus TSO working regions (in memory) are limited in size and priority. In addition, the big drawback to TSO is that it is very limited in its Input/Output support compared to CMS. TSO only supports I/O to direct access devices, not tapes or unit record equipment. This places a serious limitation on the LARSYS system, since LARSYS was designed for tapes and unit record equipment as well as direct access devices. For these reasons it was decided to convert LARSYS III.1 into a two part system. Part One is an interactive Control Card Checker implemented through TSO, and Part Two is a batch version of the LARSYS III.1 system.

The Control Card Checker is designed to allow users to interactively (at a terminal) run LARSYS control card decks through the system and perform only error checking. The card images are checked for syntax and ambiguous meaning using the same routines which LARSYS uses in the '-CHECKOUT'

mode of operation. As errors are found, the user is notified at his terminal and is asked to change the card. The replacement cards are then stored along with those with no errors to form a corrected deck. This corrected deck may then be used with the batch version of LARSYS.

It is envisioned that the user will run LARSYS jobs in both the interactive and batch steps. Although not mandatory, this procedure is recommended, since batch processing will terminate if an error is detected on a control card. With use of the Control Card Checker, however, this limitation should not be encountered very often.

The batch version of LARSYS includes virtually the entire LARSYS III.1 system, with all commands and functions implemented which are available at Purdue. Although the system is complete, the routines that dealt with Purdue's special Display Device were not changed at TAMU due to lack of this equipment. Thus this processor is not considered operational.

The major changes to LARSYS III.1 were in the interactive requests to the user and in the tape operations of the system. All interactive requests to the user have been preceded by a call to a FORTRAN routine (INTREQ) which outputs a message indicating that an interactive request was made and the name of the routine making the request. After

this message is output, execution is terminated. The Tape Operations Program (TAPOP) in LARSYS III.1 was completely re-written since this program was heavily CMS dependent. The new TAPOP handles all of the old TAPOP Entry Points and operands. It also includes two subroutines and various CMS functions from LARSYS III.1. The new TAPOP is totally transparent to the FORTRAN processing routines.

This Volume is intended to be used as an Addendum to the LARSYS User's Manuals supplied by Purdue. The remainder of this Volume is broken down into three Sections. The First Section deals with the Control Card Checker, Section Two describes the execution of the batch implementation of LARSYS III.1, and Section Three includes procedures for ERTS-MIST Conversion.

TSO Control Card Checker

Introduction. The TSO Control Card Checker is implemented under TSO and is thus subject to those limitations mentioned in the Introduction. In particular, control card decks must first be staged to a direct access device before they can be processed. This is not necessarily a disadvantage, as the full range of TSO functions are available to the user, including the 'EDIT' command whereby card images are entered and edited at a terminal and stored on disk. In short, then, the user may build his control card deck on disk using the TSO Editor, run the TSO Control Card Checker against this data set and, finally, use the TSO Command procedure 'RUNLARS' to run batch LARSYS using the corrected control card deck as input.

The remainder of this section provides a further description of the TSO Control Card Checker. It is assumed that the reader is already familiar with LARSYS and has had some exposure to the TSO command set. If the latter is not true, the IBM manuals OS/VS2 TSO Terminal User's Guide (GC28-0645-X) and OS/VS2 TSO Command Language Reference (GC28-0646-X) are recommended.

Modes of Operation. The TSO Control Card Checker is designed

to operate in one of two different modes. These are explained below with examples of use.

A. '-QUICKEDIT' Mode.

This mode is essentially the same as that provided by the LARSYS system as implemented under CMS at Purdue. Cards are only presented to the terminal if they have an error. The user then types a response which satisfies the error message and processing continues. To enter this mode a new Initialization function ('-QUICKEDIT') has been added to the LARSYS Initialization command set.

Format

-QUICKEDIT

Usage

When this function card is encountered in the input stream the remainder of the card deck is edited in the '-QUICKEDIT' mode.

Figure 2. Example 1 of Invocation of TSO Control Card Checker

```
larsys pictst
 I9001 FOLLOWING CONTROL CARD READ - ENTER DEL, CHG, OR NULL (CTLWRD)
     -QUICKEDIT

     OK....


 I0196  RUNNING CONTROL CARD CHECKOUT     (LARSMN)
 I0092 PICTUREPRINT FUNCTION REQUESTED      (PICSUP)
     HISTO COMPUTE,DOSK

  E510  ONLY ONE PARAMETER ALLOWED ON HISTOGRAM CARD - TYPE CORRECT CARD
histo disk
     SUMBYLS M,$,/,F,
  E102  INVALID KEYWORD - TYPE CORRECT CARD     (CTLWRD)
symbols m,$,/,f,
 I0237  ALL CONTROL CARDS FOR PICTUREPRINT HAVE BEEN READ     (PICRDR)
 I0093  PICTUREPRINT FUNCTION COMPLETED     (PICSUP)
 I0004  END OF INPUT DECK - RUN COMPLETED     (LARSMN)
READY
```

NOTE: For completeness, the '-QUICKEDIT' function has been added to the batch version of LARSYS. When encountered during a batch run it causes no change in processing.

B. Training Mode.

This mode was developed to aid in the training of new LARSYS users. It differs from the '-QUICKEDIT' mode in that each card image in the input control card deck is first presented to the terminal user where he is given the chance to alter it before it is sent to LARSYS error checking routines. Message I9001 is typed first, then the card image.

Figure 3. Example 2 of Invocation of TSO Control Card Checker

```
larsys pictst
DATA SET CHECKOUT.LIST NOT IN CATALOG
 I9001 FOLLOWING CONTROL CARD READ - ENTER DEL, CHG, OR NULL (CTLWRD)
     -QUICKEDIT
del
 I9001 FOLLOWING CONTROL CARD READ - ENTER DEL, CHG, OR NULL (CTLWRD)
     RUN LARSYS

     OK....

 I9001 FOLLOWING CONTROL CARD READ - ENTER DEL, CHG, OR NULL (CTLWRD)
     *PICTURE


 I0196  RUNNING CONTROL CARD CHECKOUT     (LARSMN)
 I0092 PICTUREPRINT FUNCTION REQUESTED      (PICSUP)
 I9001 FOLLOWING CONTROL CARD READ - ENTER DEL, CHG, OR NULL (CTLWRD)
     HISTO COMPUTE,DOSK
chg /compute,do/di/
 I9001 FOLLOWING CONTROL CARD READ - ENTER DEL, CHG, OR NULL (CTLWRD)
     SUMBOLS M,$,/,F;
chg /u/y/
 I9001 FOLLOWING CONTROL CARD READ.- ENTER DEL; CHG, OR NULL (CTLWRD)
     DISPLAY RUN(71053001),LINES(200,300,2)
```

Figure 3. Example 2 of Invocation of TSO Control Card Checker (continued)

```
I9001 FOLLOWING CONTROL CARD READ.- ENTER DEL, CHG, OR NULL (CTLWRD)
    CHAN 4,9,13

I9001 FOLLOWING CONTROL CARD READ.- ENTER DEL, CHG, OR NULL (CTLWRD)
    END

I0237  ALL CONTROL CARDS FOR PICTUREPRINT HAVE BEEN READ    (PICRDR)
I0093  PICTUREPRINT FUNCTION COMPLETED     (PICSUP)
I0004  END OF INPUT DECK - RUN COMPLETED     (LARSMN)
READY
```

The user then responds with a delete or change command or a null line. These responses are defined below.

1. Delete Command.

   This command gives the user the option to delete the control card. When deleted it will not be found in the rebuilt control card deck.

   Format

   del

2. Change Command.

   This command gives the user a chance to change or replace a string of characters on the given card. By using the occurrence number option, the same string may be changed a number of times.

   Format

   chg x string1 x string2 x [n]

   where x is any character which is used to delimit the strings. Obviously x should be chosen such that it will not be confused with either string1

or string2.

stringl is the string to be replaced. If stringl is empty, string2 is concatenated to the front of the existing control card.

string2 is the replacement string. If string2 is empty, then stringl is deleted from the control card.

n is the occurrence number which expresses the number of occurrences of stringl to be replaced by string2. It must be exact, i.e., if it is desired to change all occurrences of a string and that string occurs 10 times then n must be 10. The default for n if it is omitted is 1.

Examples

Original String - ABCDEF
Response - chg /CD/NP/
New String - ABNPEF

Original String - ABCABCD
Response - chg /CD/CXYZD/
New String - ABCABCXYZD

Original String - 23.5/8.6/12.2/5.0

Response - chg ?/8.6??

New String - 23.5/12.2/5.0

Original String - ZYXOPQ

Response - chg //ABC/

New String - ABCZYXOPQ

Original String - *1*2*3*4*5*6

Response - chg /*/-/4

New String - -1-2-3-4*5*6

3. Null Line.

A null line is transmitted by simply pressing the new line button on the terminal, e.g., on IBM 2741 type terminals this is the carriage return. This indicates to the system that the displayed control card is to be accepted as it stands.

Format

(New Line)

After the above exchange is completed between the system and the user, the control card, unless deleted, is sent to the regular LARSYS editing routines. As in the '-QUICKEDIT' mode, if an error is encountered the appropriate LARSYS error message will be generated.

Differences in Command Processing. In general, the interactive checkout system will syntax check each of the LARSYS commands and functions. The three levels of control are discussed in the LARSYS User's Manual, pages 2-33 - 2-49. Each is presented below with a discussion of their implementation under the interactive checkout system.

A. Control Commands.

All of those commands which are normally entered as CMS-type Executive Commands may be entered on a card image to the interactive checkout routines. Each command is checked for correct spelling and operands, if any. In addition, some of the commands are actually processed and cause output to appear at the terminal. Table 5 is a list of these commands and the action taken for each.

B. Initialization Commands.

All of the Initialization functions are processed and executed as in the original version of LARSYS. The TSO Control Card Checker assumes the presence of a '-CHECKOUT' card whether one is there or not. This allows the user to use a set of checked control cards directly as the input to the batch LARSYS system.

C. Processing Commands.

The processing function cards and their associated

Table 5 . TSO Control Card Checker Action on LARSYS Control Commands

Command	Action Taken Other Than Syntax Check
batch	None
begin	None
ccinput	None
clear	None
disconnect	None
histdeck	None
i larsys	None
list [operand]	Provides the appropriate Index list at the terminal.
login	None
msg	None
news [operand]	Provides the appropriate News file at the terminal.
print [operand]	None
punch [operand]	None
quit	Stops interactive processing; Returns user to TSO environment without re-building control card data set.
reference [operand]	Provides the appropriate Reference file at the terminal.
run larsys	None
statdeck	None
stop	None
suspend	None
termtest 'nn'	Not supported, warning message printed.

modifier cards are thoroughly checked for errors using the LARSYS processing/error checking routines. As in the original version, the errors which can be caught only by actually processing data are not flagged in the interactive system.

Examples of Use. The intent of this section is to give graphic examples as to how the user might use TSO and the TSO Control Card Checker to make a LARSYS run. The procedure is broken into three simple steps.

1. Using the TSO editor, build the set of control cards needed to make the run (Figure 4).

   Note that this example data set has built-in typographical errors so that it can be used in the Quickedit and Training examples.

Figure 4. Example of Building LARSYS Control Cards Through TSO

```
edit pictst new data
INPUT
00010-quickedit
00020run larsys
00030*picture
00040histo compute,dosk
00050sumbyls m,$,/,f,
00060display run(71053001),lines(200,300,2)
00070chan 4,9,13
00080end
00090
EDIT
save
EDIT
end
READY
```

2. Invoke the TSO Control Card Checker, indicating with a parameter the name of the above control card data set (Figures 2, 3).

3. Submit to the batch processor a LARSYS batch job using the RUNLARS Procedure (Figure 5).

Figure 5. Example of Submitting a TSO Control Card Data Set to Batch LARSYS

```
exec runlars 'pictst1'
JOB LARSYS1  SUBMITTED
READY
```

TSO Control Card Checker Error Messages. The following messages have been added to those which may be received at the user's terminal.

Table 6. TSO Control Card Checker Error Messages

Message - I9001 FOLLOWING CONTROL CARD READ - ENTER DEL, CHG OR NULL (CTLWRD)

Error - None

Response - Enter one of the listed choices (See discussion of Training Mode.)

Message - E9001 STRING TO CHANGE NOT FOUND ON CONTROL CARD - TRY AGAIN (CTLWRD)

Error - In the Training Mode, a CHG command has been entered. The 'stringl' string was not on the original control card.

Response - Enter new change card or DEL or null line.

Table 6. TSO Control Card Checker Error Messages (continued)

Message - E9002 REPLACEMENT STRING MAKES CARD IMAGE TOO LONG - TRY AGAIN (CTLWRD)

Error - In the Training Mode, a CHG command has been entered. The resultant string after the replacement of 'string1' with 'string2' is greater than a card image (80 characters).

Response - Enter new change card or DEL or null line.

Message - E9003 SYNTAX ERROR ON CHANGE REQUEST - TRY AGAIN (CTLWRD).

Error - In the Training Mode, a CHG command has been entered. A syntax error has occurred on the command after the command name, i.e. after CHG

Response - Check the previous response, comparing with the examples in this text. Enter new change command or DEL or null line.

Message - E9004 INVALID RESPONSE TO MSG I9001 - TRY AGAIN (CTLWRD)

Error - In the Training Mode, something besides CHG, DEL or null line has been entered.

Response - Enter correct command.

Data Sets used in the TSO Control Card Checker Program.

The data sets listed below with their FORTRAN Data Set Reference Numbers (DSRN) are those required to run the TSO Control Card Checker. The list is presented here only in the interest of the TSO user who would like to see which data sets he has allocated. They are described either in the LARSYS Sys-

tem Manual or the accompanying TAMU LARSYS System Programmer Manual.

Table 7. Data Sets Used in the TSO Control Card Checker Program

DSRN	File
1	Statistics File
5	Control Card Input
6	Printer Output (Not applicable)
8	Error Message File
15	Terminal Keyboard (Input)
16	Terminal Typewriter (Output)
25	Temporary card image data set to collect corrected control cards.
31	Permanent STATISTICS File (Not used)
32	Permanent HISTOGRAM File (Not used)
33	LARSYS INDEX File
34	PRIORITY NEWS File
35	SYSTEM NEWS File
36	SCHEDULE NEWS File
37	SYSTEM UPDATE NEWS File
38	Control Commands Reference File
39	Initialization Commands Reference File
40	Processor Commands Reference File
41	Runtable Reference File

TAMU Batch LARSYS Execution

The TAMU batch version of LARSYS is a virtually complete implementation of Purdue's LARSYS III.1. All functions and processors have been thoroughly tested, except those which deal with Purdue's special Display Device.

Executing TAMU Batch LARSYS. The LARSYS user has two distinct "paths" to the batch LARSYS version. The first path is explained in the preceding section, the TSO Control Card Checker. The user submits his LARSYS control card deck to batch LARSYS via the TSO Command Procedure 'RUNLARS'. This path should be used anytime a new control card deck is being run. It would be to the user's advantage to use the TSO Control Card Checker as often as possible, just to be assured that a syntax error has not been made while keying in the LARSYS commands.

The second path to batch LARSYS is through execution of the OS Command Procedure 'LARSYS'. This Procedure invokes the LARSYS system and contains the Job Control Language statements necessary for LARSYS execution. This path should only be used when the user is sure that the control cards contain no errors.

The Procedure 'LARSYS' has two parameters, HIST and STAT. These parameters are optional and, if present, are

specified on the 'EXEC' card. They are optional only in the sense that if a Statistics or Histogram deck need not be permanently saved, then these parameters are not required. On the other hand, if either of these decks needs to be saved, then the appropriate parameter must be specified. These parameters specify a private catalogued data set, which overrides the standard LARSYS data set. This feature corresponds to the use of the P-Disk in Purdue's LARSYS.

LARSYS Interactivity Status. The major difference between batch LARSYS and Purdue's LARSYS is that the interactivity has been taken out of LARSYS for the batch version. The user must keep this fact in mind at all times. It was discovered during testing of LARSYS that about 90% of all interactivity was due to syntax errors in control cards. Hopefully, these errors will now be caught prior to batch LARSYS execution via the TSO Control Card Checker. The other 10% of the interactivity, though, now becomes the user's responsibility. For example, the user should be sure that the Run he requests is in the System Runtable prior to running LARSYS. The TSO Control Card Checker cannot check for errors of this sort. Previously, LARSYS would ask for an interactive input for this situation, but this is not possible in batch LARSYS. Another example would be the use of the 'OPTIONS TYPE' card when running the SEPARABILITY Processor. The 'OPTIONS TYPE' card allows interactive user

input, which cannot be handled by batch LARSYS. The loss of interactivity in batch LARSYS does not cause any reduction in processing capabilities, only a minor change in the user's method of using LARSYS.

Batch LARSYS Control Cards. The control cards for LARSYS processing essentially remain unchanged between Purdue's LARSYS III.1 and the TAMU batch LARSYS. This includes all Initialization and Processor commands as well as the Control commands that were handled by CMS EXEC modules, i.e., BATCH, LIST STATISTICS, etc. These Control commands are now executable in card form, although only specific ones have any meaning, since they were originally of an interactive nature. Table 8 shows the batch LARSYS execution changes for those commands which have been altered.

Batch LARSYS Control Card Examples. Some examples of deck set-ups for typical batch LARSYS jobs are listed below.

1.
```
// JOB
/*JOBPARM REGION=384,TAPE9=3
// EXEC LARSYS,STAT='USER.LIB1',HIST='USER.LIB2'
//SYSIN DD *
RUN LARSYS
*CLASSIFYPOINTS
                } LARSYS Processor Control Cards
/*END
```

The 'JOBPARM' card above indicates that LARSYS runs in 384K bytes of core memory and requires a maximum of three 9-track tapes. The 'EXEC' card shows that the user

Table 8. TAMU Batch LARSYS Command Alterations

Command Type	Command	Action Taken by Batch LARSYS
Control	BATCH	Warning message. Execution continues.
	BEGIN	Warning message. Execution continues.
	CCINPUT	Executable for 'CARDS' option only. Otherwise execution terminated.
	CLEAR	As documented in the LARSYS User's Manual.
	DISCONNECT	Warning message. Execution continues.
	HISTDECK	As documented in the LARSYS User's Manual.
	I LARSYS	Warning message. Execution continues.
	LIST	As documented in the LARSYS User's Manual.
	LOGIN	Warning message. Execution continues.
	MSG	Warning message. Execution continues.
	NEWS	As documented in the LARSYS User's Manual.
	PRINT	Warning message. Execution continues.
	PUNCH	Warning message. Execution continues.
	QUIT	Execution terminated.
	REFERENCE	As documented in the LARSYS User's Manual.
	RUN	Mandatory. Executable for 'LARSYS' option only.
	STATDECK	As documented in the LARSYS User's Manual.

Table 8. TAMU Batch LARSYS Command Alterations (continued)

Command Type	Command	Action Taken by Batch LARSYS
	STOP	Execution terminated.
	SUSPEND	Error message. Execution terminated.
	TERMTEST	Error message. Execution terminated.
Initialization	-QUICKEDIT	None. For TSO Control Card Checker only.
	The remainder of the Initialization commands are as documented in the LARSYS User's Manual.	
Processor	All	As documented in the LARSYS User's Manual.

will process his STAT deck and HIST deck, through the proper control commands, in conjunction with USER.LIB1 and USER.LIB2, both of which are private catalogued data sets. Input is from the card reader as defined by the 'SYSIN DD' card.

2.
```
// JOB
/*JOBPARM REGION=384,TAPE9=3
// EXEC LARSYS,HIST='USER.HIST.DATA'
//SYSIN DD DSN=USER.LARSYS.INPUT.DATA,DISP=SHR
/*END
```

The 'EXEC' card shows that the user will process his HIST deck in conjunction with USER.HIST.DATA, a private catalogued data set. Also, the lack of the STAT parameter on the 'EXEC' card indicates that the user has no need of a permanent STAT deck file, either to be used or saved. The 'SYSIN DD' card indicates that LARSYS input will be from the private catalogued data set USER.LARSYS.INPUT.DATA.

3.
```
// JOB
/*JOBPARM REGION=384,TAPE9=3
// EXEC LARSYS
//SYSIN DD *
```
} LARSYS Control Cards
```
/*END
```

This example is given to show that neither the STAT nor the HIST parameters need be specified if the user does not want to access or save his STAT or HIST decks.

ERTS-to-MIST Conversion

The ERTS-to-MIST Conversion program, EMCNV, operates on the standard four file ERTS data set to produce the LARSYS input file, the Multispectral Image Storage Tape, or MIST tape. This section documents the use of EMCNV.

The ERTS Input Data Set

ERTS Data Set Forms. The input data set to EMCNV may consist of one of three forms of the standard ERTS data set. The first of these forms, that in which the majority of the past ERTS data has been released, consists of four tapes, each of which contains one 25 x 100 nautical mile section of the ERTS scene. All four of these tapes, in addition to the tape to be MIST formatted, must be mounted and attached to the computer during the execution of EMCNV.

The second ERTS data set form accepted by EMCNV is the two tape, two file per tape data form. The difference between this form and the four tape form is that two data files from two tapes have been combined onto one tape. Thus, in the two tape data set form, only three tapes must be accessible to EMCNV.

The third possible form of the ERTS data is the one tape, four file per tape form. In this form, all four of the ERTS data files are on one tape. Thus, for one tape ERTS data sets, only two tapes must be available to EMCNV.

Due to the necessity of tape-to-disk copy operations for the two tape and one tape ERTS data set forms, it is suggested that four tape ERTS data sets be used as much as possible for economical reasons.

Input Error Recovery Procedures. All of the Input/Output operations in EMCNV are in FORTRAN; thus, any I/O errors are processed by the error handling facilities of the Operating System.

The three errors most likely to occur are data transmission errors, a premature File Mark on one or more of the ERTS files, or an unusable segment of an ERTS scan line. In all cases, the spectral reflectances for the segment of the ERTS scan line in error are set to zero. In the case of a premature File Mark on an ERTS file, that segment of all of the following ERTS scan lines is also set to zero. Thus, a complete MIST tape containing the standard number of 2340 scan lines is always generated, even though some of the data may be zero.

In the event that any of the above errors occur in EMCNV, appropriate output messages are generated to indicate to the user the unusable ERTS scene sub-areas. See Figure 6.

Figure 6. Example of ERTS-to-MIST Conversion Error Output

ERTS INPUT TAPE ERRORS -

MODULE	* - ERTS -* TAPE	FILE	ERTS/MIST SCAN LINE	* ----- ERROR DATA ERROR -----	TAPE CHECK -----	FILE MARK ---- *
MAIN	2	1	10	X		
MAIN	4	1	10		X	
MAIN	1	1	15		X	
MAIN	3	1	20			X
MAIN	1	1	25	X		
MAIN	2	1	35		X	
MAIN	1	1	40	X		
MAIN	2	1	45		X	
MAIN	1	1	50		X	
MAIN	4	1	50	X		

The MIST Output Data Set

The MIST data set output from EMCNV is composed of two files on tape. The first file is the actual MIST data file and contains the Header record and the spectral Data records. The second file is a dummy MIST file and is comprised of only a Header record containing zeros. This dummy file is necessary in the operation of LARSYS tape copy and error detection functions.

Further information concerning MIST output files can be found in the LARSYS User's Manual, Appendix IV, June 1, 1973, and the LARSYS System Manual, June 1, 1973, pages 5-52 through 5-58.

Conversion Execution

EMCNV is currently implemented within an OS Command Procedure called EMCNV. The Job Control Language in the Procedure EMCNV is set to process a four tape ERTS data set in standard form. The discussions below present the instructions necessary to execute the standard conversion operation, alterations to the JCL necessary to convert ERTS data in one of the other two input forms and input data card formats and contents.

Standard Conversion. The JCL statements below invoke the Procedure EMCNV.

```
//JOB
/*JOBPARM     REGION=128,TAPE9=5
//EMCNV EXEC EMCNV,ERTS1='T1',[1]ERTS2='T2',[1]ERTS3='T3',[1]ERTS4='T4',[1]
// MIST='MT'[2]
//SYSIN       DD  *
Data Cards
/*
```

---

[1] Alphanumeric installation numbers of ERTS tapes 1-4. The sequence of occurrence of these tapes is immaterial.

[2] Alphanumeric installation number of MIST tape.

Non-Standard Conversion. For two and one tape multi-file ERTS data sets, the JCL in the Procedure EMCNV must be overridden in the following fashions.

For two tape ERTS data sets, execute

```
//JOB
/*JOBPARM  REGION=128,TAPE9=3

//EMCNV EXEC EMCNV,ERTS1='T1',ERTS2='T2',MIST='MT'

//FT01F002 DD UNIT=TAPE9,VOL=SER=&ERTS1,LABEL=(2,NL,,IN),
//         DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=3500,OPTCD=Z)

//FT02F002 DD UNIT=TAPE9,VOL=SER=&ERTS2,LABEL=(2,NL,,IN),
//         DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=3500,OPTCD=Z)

//FT03F001 DD  DUMMY

//FT04F001 DD  DUMMY

//SYSIN    DD  *

Data Cards

/*
```

For one tape ERTS data sets, execute

```
//JOB
/*JOBPARM    REGION=128,TAPE9=2

//EMCNV EXEC EMCNV,ERTS1='T 1',MIST='MT'
//FT01F002  DD UNIT=TAPE9,VOL=SER=&ERTS1,LABEL=(2,NL,,IN),
//          DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=3500,OPTCD=Z)

//FT01F003  DD UNIT=TAPE9,VOL=SER=&ERTS1,LABEL=(3,NL,,IN),
//          DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=3500,OPTCD=Z)

//FT01F004  DD UNIT=TAPE9,VOL=SER=&ERTS1,LABEL=(4,NL,,IN),
//          DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=3500,OPTCD=Z)

//FT02F001  DD  DUMMY

//FT03F001  DD  DUMMY

//FT04F001  DD  DUMMY

//SYSIN     DD  *

Data Cards

/*
```

EMCNV Card Input Conversion Parameters. There are two conversion parameter data cards to be input to EMCNV. Their formats and contents are shown in Table 9.

Table 9. ERTS-to-MIST Conversion Parameter Input Formats

Card	Col	Format	Contains
1	1-8	2A4	Installation Number of First ERTS Tape
	11-18	2A4	Installation Number of Second ERTS Tape
	21-28	2A4	Installation Number of Third ERTS Tape
	31-38	2A4	Installation Number of Fourth ERTS Tape
	41-50	I10	Number of ERTS Tapes (D=4)[1]

---

[1] Default values.

Table 9. ERTS-to-MIST Conversion Parameter Input Formats (continued)

Card	Col	Format	Contains
2	1-10	I10	MIST Tape Number
	11-20	I10	File Number on MIST Tape
	21-30	I10	MIST Run Number
	31-40	I10	Continuation Code (1 if MIST Tape Continuation; 0 Otherwise.)
	51-55	A4,A1	State Name or ERTS Scene User Identication.
	61-72	3A4	Conversion Date - Form=MMM DD, YYYY.

Conversion Error Processing. EMCNV performs several edit functions through the initial phase of execution to insure that a valid MIST tape will be produced. Table 10 contains a list of the possible errors and their interpretations. Note that any error occurrence is fatal to EMCNV execution.

Conversion Output. Figure 7 shows the printed output produced after a successful conversion. The spectral band parameters and calibration constants are referencible in the ERTS Data User's Guide available from Goddard Spaceflight Center, Greenbelt, Maryland.

Table 10. ERTS-to-MIST Conversion Execution Errors

Module	Error	Interpretation	Indicative Parameters Printed
MAIN	MN01	Wrong MIST tape mounted.	Tape specified/Tape mounted.
	MN02	File number error on MIST tape.	File number in MIST Header record/ Actual file number.
	MN03	File Mark hit reading MIST Header record.	File number.
	MN04	Wrong MIST file specified.	File specified/First empty file on MIST tape.
CHKIDA	CH01	Wrong file number in ERTS ID record (1-4).	Tape number being processed/File number being processed.
	CH02	Wrong number of files in ERTS ID record (4).	Tape number being processed/File number being processed.
	CH03	Duplicated ERTS file. File already found.	Tape number being processed/File number being processed.
	CH04	Number of tapes not equal to 1,2, or 4.	Number of tapes.
	CH05	Unequal ERTS ID records.(File 1-4).	ERTS file number/ERTS ID record error word.
	CH06	Unequal ERTS Annotation records. (File 1-4).	ERTS file number/ERTS Annotation record error word.
	CH07	ERTS Data record bytes not word multiple.	Number of ERTS Data record bytes in fourth file processed.
	CH08	File Mark hit reading ERTS ID record.	Tape number being processed/File number being processed.
	CH09	File Mark hit reading ERTS Annotation record.	Tape number being processed/File number being processed.
ANSPLT	AN01	Wrong month in ERTS Annotation record.	None.

Figure 7. Example of ERTS-to-MIST Conversion Normal Output

ERTS/MIST CONVERSION - TEXAS A&M UNIVERSITY -

ERTS PARAMETERS

ERTS TAPES	FLIGHT LINE IDENTIFICATION	DATE MO/DY/YR/HOUR	SENSOR ALTITUDE	SENSOR HEADING
ZZ3653 ZZ3652 ZZ3650 ZZ3651	11531710500TEXAS	12/23/72/1010	3062000	190

MIST PARAMETERS - CONVERTED DEC 6, 1975

MIST TAPE	FILE	RUN NUMBER	CC	NUMBER OF LINES	NUMBER OF CHANNELS	NUMBER OF SAMPLES/CHANNEL/LINE
989	1	72063700	0	5	4	3232

SPECTRAL DATA

CHANNEL	LOW MCM	HIGH MCM	SUGGESTED CAL PULSE (HEADER RECORD) C0	C1	C2	* -- CALIBRATION VALUES - * (DATA RECORDS) C0	VC0	C1	VC1	C2	VC2
1	0.5	0.6	0.0	2.48	0.0	0	0	128	0	0	0
2	0.6	0.7	0.0	2.00	0.0	0	0	128	0	0	0
3	0.7	0.8	0.0	1.76	0.0	0	0	128	0	0	0
4	0.8	1.1	0.0	4.60	0.0	0	0	64	0	0	0